ユーザのセキュリティ意識向上を目的としたパケットヘッダ情報可視化システム

中野 翔太* 白井 治彦*** 高橋 勇** 黒岩 丈介** 小高 知宏* 小倉 久和**

A Proposal on End-User Network Security System To Visualize Packet Header Information

Shouta NAKANO *, Haruhiko SHIRAI*, Isamu TAKAHASHI**, Jousuke KUROIWA****
Tomohiro ODAKA and Hisakazu OGURA****

(Received January 31, 2007)

To Improve security awareness of end-user, we designed and implemented the visualization system of packet header that represents the condition of network communication. We implemented our system using Microsoft Visual Studio 2005 with Winpcap library and the development language of system was Visual C++. This system captured a packet and visualized flow of network traffic, protocol and the time it captured a packet.

Key Words : Network, Security, Visualization, Packet Header

1. はじめに

近年，情報通信技術の発展に伴い，ネットワーク通信（以下，通信）の高速化・大容量化が進んでいる．しかし，技術の進歩における弊害が表面化しているのもまた事実である．それは，通信というものが目に見えず抽象的な存在であるため，一般的なユーザにとっては実際に通信がどのような仕組みで，どのようなことが行われているかがわからないという問題である．この問題の一例としてファイル共有ソフトの利用が引き起こした企業や官公庁による一連の内部情報流出事件が挙げられる．これは通信に対する意識の低さが引き起こした事件と言える．このような問題を解決するためには，ユーザがセキュリティ意識を高くもつ必要がある．そのためには，通信についての知識や構造を知る必要がある．そのためのツールも数多く存在するが，例えばプロトコルの名称やパケットの構造などネットワークの専門的な知識がないとわからないことも数多く存在し，一般的なユーザには敷居が高いものと考える．また，ユーザに通信に対して興味を持ってもらわなければユーザは通信に関する知識を身につけようとはしないと考える．

そこで本研究では，ネットワークに対する知識が乏しいエンドユーザを対象とし，ユーザが使用しているコンピュータにおけるネットワークの情報をわかりやすく提示するシステムを提案する．このシステムを利用することで，ユーザは抽象的な存在であるネットワークの世界に興味をいだくきっかけをつくり，その興味をユーザのセキュリティ意識向上へとつなげることが本研究の目的である．システムにおけるわかりやすさという視点に着目し，通信の状況をできるだけシンプルに表示することが重要だと考えた．そこで，利用中のコンピュータがやりとりするパケットにおけるヘッダ情報の一部のみを表示することで情報を簡素化することとした．ヘッダ情報はネットワークの流れをわかりやすく理解できる

* 工学研究科原子力・エネルギー安全工学専攻
** 工学研究科知能システム工学専攻
*** 技術部
* Nuclear Power and Energy Safety Engineering Course, Graduate School of Engineering
** Human and Artificial Intelligent Systems Course, Graduate School of Engineering
*** Dept. of applied Physics

ものだけを選択してユーザに提示するものとした．具体的には，通信におけるデータ量を表すパケットのサイズ，通信が行われているサービスを表す宛先ポート番号というヘッダ情報をシステムで利用する．システムはネットワークから取得したパケットからヘッダ情報のみを利用し，そこから前述の 2 つのヘッダ情報に加えて，不必要なパケットをフィルタリングするためのヘッダ情報の抽出・保存・処理を行う．そして，単位時間ごとの宛先ポート別通信量上位 3 位をグラフ化して表示することでわかりやすいインターフェースを実現した．システムは，一般的に利用者が多い OS であること，そして対象とするユーザを考慮に入れた結果，Windows 系 OS で動作するものとした．

本論文は 2 章でセキュリティの現状とそれをふまえてのシステムの設計について，3 章ではパケット可視化システムの実装について，4 章では動作実験について実際のインターフェース画面を交えて述べ，5 章では考察，そして 6 章で今後の課題について述べる．

2. セキュリティの現状とシステムの設計方針

2.1 セキュリティの現状

常時接続形態のネットワークの普及でコンピュータがネットワークに接続する時間が長くなったことにより通信量が増大し，コンピュータウィルスやワームなどの悪意あるプログラムによる不正アクセスの危険性が高くなっている．同時にクラッカーによる人為的な不正アクセスの危険性も高くなっている．このような悪意あるプログラムやユーザはコンピュータ内のデータを破壊したり，コンピュータそのものを停止させたりといったユーザにとって不利益な活動を行う．

しかし，このような危険な状況にも関わらずウィルス対策ソフトやファイアウォール等の導入といった防御策を全く行わずに通信を利用するユーザが後を絶たないという現状がある[1]．これがひとつめの常時接続環境における弊害である．この状況は，ネットワーク環境の改善によりインターネット利用者の間口が広がり，誰でも通信が利用できるようになったことが一因であると考えられる．

ふたつめの弊害は，ファイル共有ソフトの利用による企業や官公庁などの一連の内部情報漏洩事件である．これらの事件はファイル共有ソフト利用者を標的としたワーム型ウィルスに感染することが原因である．感染するとウィルスがコンピュータ内の個人情報や企業の情報をネットワークに流出させてしまう．

表 1　近年のネットワーク環境，常時接続環境の弊害とその原因

事象	概要
近年のネットワーク環境	大容量常時接続形態
常時接続環境における弊害	不正アクセスの危険性の増大とそれに対するユーザの無関心
	ファイル共有ソフト利用によるウィルス感染
常時接続環境における弊害の原因	エンドユーザがネットワークに興味を持たない
	ネットワーク・通信の構造を知らない

ファイル共有ソフトの使用自体に根本的な問題があるようにも思われるが，不正アクセスに対して無防備なユーザの問題も含めて，これらの弊害にはある共通した問題があるように思われる．それはインターネットを利用する一般的なユーザが，ネットワークがどのような構造であるか，そして通信がどのようにして行われているかを知らないこと，またネットワークそのものに対して興味を持たないことである(表 1 参照)．

その理由のひとつとしてはネットワークが目に見えず，抽象的な存在のためであると考える．これは通信が不透明であり，かつネットワークにおけるデータのやりとりが複雑な構造をしていることに起因していると考えられる．もうひとつの理由は通信が常時接続形態になったことで通信を利用したアプリケーションが増えていることであると考える．

そのため，ブラウザやメーラーのようにユーザによる操作によって通信を行うアプリケーションだけでなく，ユーザが意図しないところで通信を行うアプリケーションが非常に増えている．例えば，ウィルス対策ソフトに代表される定期的なアップデートが必要なソフトウェアはユーザが意図しないところで通信が行われてしまうアプリケーションである．

このような理由から，ネットワーク環境の改善によりユーザと通信との距離は縮まったが，ユーザ全体のネットワークに対する興味や理解度は平均して下がっているものと考えられる．

2.2 システムの設計方針

図 1 に 2.1 章で述べた問題に対応したシステムの設計方針を示した．本システムはネットワークにおけるエンドユーザにセキュリティ意識を高く持ってリティ意識の低いユーザが本システムの対象となるわけであるが，そのようなユーザとはすなわちイン

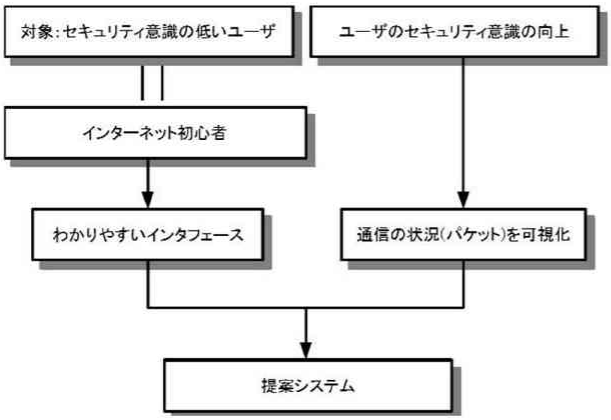

図 1　システムの設計方針

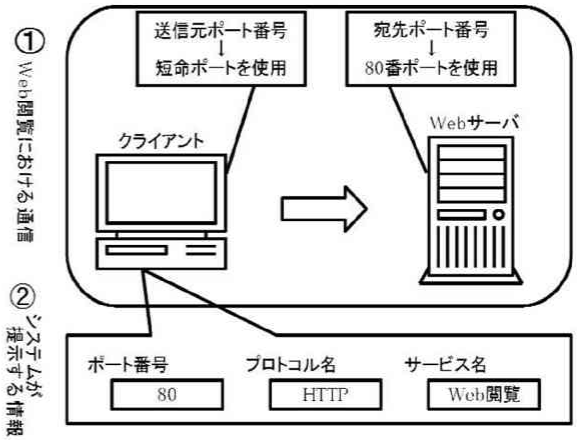

図 2　システムが提示する情報（Web 閲覧を行う場合）

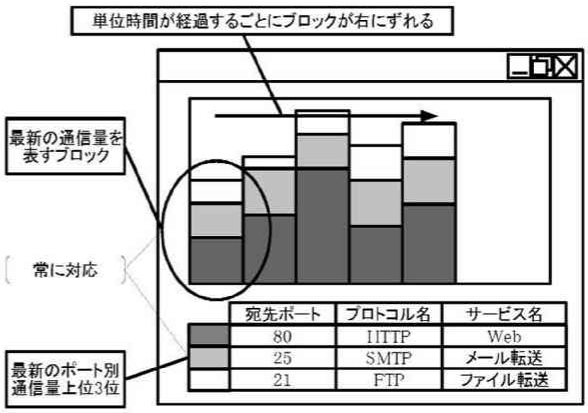

図 3　システムのインターフェース画面例

ターネット利用歴の浅いユーザである[1]．インターネットの利用歴の浅いユーザはネットワークに関する事象の認知度も低いものと考えられる．そのため，このようなユーザにセキュリティ意識を高く持ってもらうためにはなるべくわかりやすいインターフェースを持ったシステムが必要であると考える．それと同時に，このようなユーザの多くは UNIX 系 OS の利用をすることはほぼ皆無と考えられる．そこで，本システムは Windows 上でのみ動作するものとする．

また，何らかの形で通常は見ることができないネットワークにおけるデータの流れが見えるようにすることがセキュリティ意識の向上には必要であると考える[2]～[6]．つまり，通信の状況をユーザに提示するという意図をもったパケットの可視化である．

以上より，本システムは対象とするインターネットの利用歴の浅いユーザに対してシンプルでわかりやすいインターフェースでパケットを可視化するシステムとする．そのため，ユーザに提示する情報をヘッダ情報の一部のみに簡素化し，そのうえで通信の状況がわかるように設計した．システムでは以下のような情報をユーザに提示するものとする．

- 宛先ポート番号とそのサービス名
- 単位時間あたりの宛先ポート番号別通信量

まず，宛先ポート番号をユーザに提示する理由は一般的なクライアントーサーバ間の通信において，その通信を特定するのがサーバにおけるサービスだからである．クライアント側ではネットワーク上でデータがやり取りされる際に 1025 番以降の短命ポートが使われるケースがあり，その状況をユーザに把握させたいという理由から表示するものとした．例えば，クライアントが Web 閲覧を行う場合，通信を特定するのはサーバにおけるサービスであるので，クライアントが Web サーバに対して通信をすることになる．よって通信の方向としてはクライアントからサーバという方向となる(実際の通信では，クライアントーサーバ間では双方向の通信が行われる)．そのとき，クライアント側には送信元ポート番号，サーバ側には宛先ポート番号が割り当てられる．クライアント側では任意の短命ポート番号が使用され，サーバ側は HTTP(HyperText Transfer Protocol) によって使用される 80 番ポートがこの場合には当てはまる(図 2①参照)．

サービス名はヘッダ情報には含まれないが，宛先ポート番号のみの表示では対象とするユーザがどのプロトコルを使用し，どのようなサービスが行われているのかがわかりづらいと考えたため，ポート番号と対応付けて表示するものとした．前述と同様に HTTP を例にあげると，「80－HTTP－Web 閲覧」といった表示がシステムからクライアント側，すなわちユーザになされる(図 2②参照)．

また，単位時間あたりの宛先ポート番号別通信量の上位 3 位までをグラフ化しユーザに表示する．こうすることで通常は見えない通信の状況をわかりやすく見ることが可能となる．

図 3 に示したように，以上の 2 つの要素を融合した形をもってユーザに提示する．グラフ画像は 3 つのブロックからなる 1 本の縦に長いブロックが，ある単位時間の通信量の合計を表しており，
そのブロックの下から順に通信量が多いブロックである．単位時間が経過するごとに，ブロックは右方向にずれていき，グラフの一番左にあるブロックが最新の通信量を表すブロックとなる．画面下の宛先ポート番号・プロトコル名・サービス名の情報は単位時間が経過するごとに更新されていく．つまり，情報の表示はグラフ画像の一番左のブロックの情報をユーザに提示している．図 3 は最新の単位時間あたりにおいて宛先ポート別通信量の上位 3 位が HTTP，メール配送・転送に利用される SMTP(Simple Mail Transfer Protocol)，ファイル転送に利用される FTP(File Transfer Protocol)であった場合にユーザに提示するインターフェース画面を表したものである．

3. システムの構築

3.1 システムの概要

本システムはユーザに対するネットワークセキュリティ意識向上を目指したパケット可視化システムである．このシステムの構成を図 4 に示した．本システムはパケットヘッダ取得機構，パケットヘッダ処理機構，データ保存機構，表示機構という 4 つの機構から構成される．

3.2 システムの実装

2.2 章で述べたシステムの設計方針に基づき，開発環境として Microsoft Visual Studio 2005 を用いてシステムを実装した．開発言語は Visual C++を使用した．以下に，システムにおける各々の機構の詳細について示す．

3.2.1 パケットヘッダ取得機構

パケットヘッダ取得機構によってパケットにおけるヘッダ部分のみを取得する．パケットヘッダの取得には，WinPcap ライブラリ[7]を使用する．WinPcap ライブラリとは，Windows 環境においてネットワークアダプタでやりとりされるパケットの取得に関する様々な操作を行うことができるライブラリである．パケットを取得すると，格納部において WinPcap ライブラリで用意されている構造体に全ヘッダ情報が格納される．システムが動作しているときはパケットの取得と取得待機が繰り返されるので，パケットを取得する度に，状態遷移部にて取得状態・待機状態の状態遷移が絶えず行われる．つまり，パケットが全くこない場合は待機状態が続くことになる．また，時刻登録部にてパケットヘッダを取得した時刻を登録しておく．まとめると，この機構では，図 5 のようなループがシステムを終了するまで繰り返されることになる．

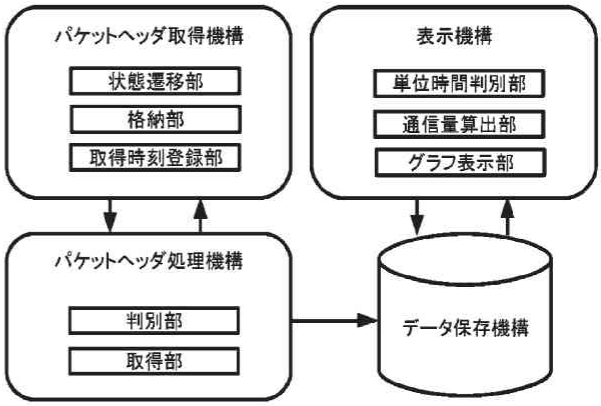

図 4　システムの構成

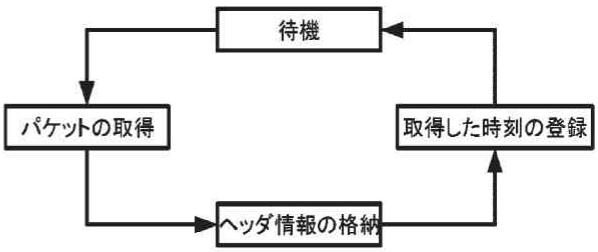

図 5　パケットヘッダ取得機構での内部構造

3.2.2 パケットヘッダ処理機構

パケットヘッダ処理機構の構造を図 6 に示した．点線で囲まれた部分がパケットヘッダ処理機構である．また，点線の内部にある図はパケットの構造を示したものである[8]．実際にはパケットの情報はパケットヘッダ取得機構にて構造体に格納されている．

まず，パケットヘッダ取得機構から渡されたヘッダ情報を判別部においてヘッダ情報を用いた選別を行う．Type Field は Ethernet における上位プロトコルが何かを示しているものであり，これの違いによりプロトコルが指定されている．本システムでは IP パケットのみを扱うため，Type Field の判別を行う．

Type Field は Ethernet フレームのヘッダに含まれるため，ここからデータを取得し判別する．Type Field が IP パケット以外のものであった場合は，処理はパケットヘッダ取得機構に戻り，待機状態とな

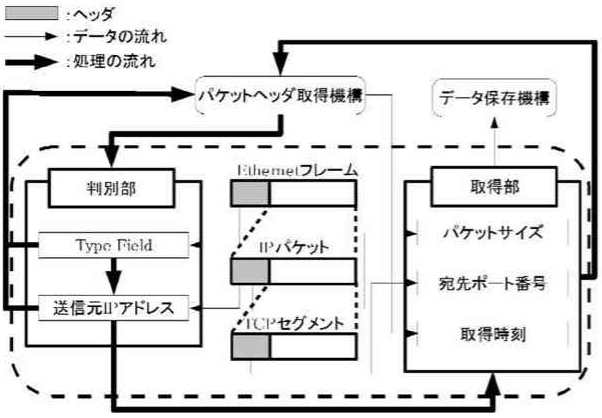

図 6　パケットヘッダ処理機構の構成

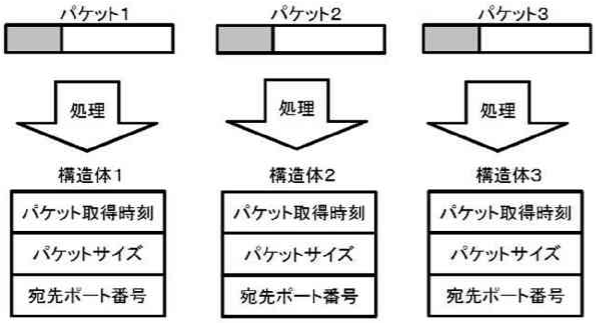

図 7　パケットヘッダ処理後に構造体に格納する情報

る．IP パケットのものであった場合は，送信元 IP アドレスの判別に処理が移る．送信元 IP アドレスの判別は WinPcap ライブラリの性質上，ネットワークアダプタにおいてやり取りされる全パケットを取得してしまうため，自分のコンピュータとは関係のないパケットが含まれてしまう．そこで，自分のコンピュータにおいて割り振られている送信元 IP アドレスによるフィルタリングを行うことで，パケットの選別を行う．送信元 IP アドレスは IP パケットのヘッダに含まれるため，ここからデータを取得し判別する．先程と同様に，送信元 IP アドレスが自分のコンピュータ以外のものであった場合，処理がパケットヘッダ取得機構に戻り，待機状態となる．ここまでの判別が終了すると，処理は判別部から取得部に移る．

取得部では，主にパケットヘッダ取得機構で格納された構造体から，本システムで必要となる情報を取得する．パケットサイズは IP パケットのヘッダから，宛先ポート番号は TCP セグメントのヘッダから，そして取得時刻はパケットヘッダ取得機構の取得時刻登録部からそれぞれの情報を取得する．取得時刻，

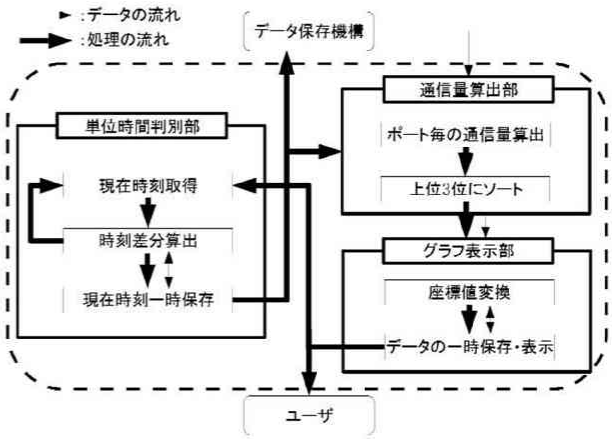

図 8　表示機構の構成

宛先ポート番号，パケットサイズの 3 つの情報がデータ保存機構に渡される．取得部における一連の情報の取得が終了すると，処理はパケットヘッダ取得機構に戻り，待機状態となる．

3.2.3　データ保存機構

データ保存機構では，パケットヘッダ処理機構から受けたデータを本システム実装のために用意した構造体に格納する．実際に構造体に格納する情報は図 7 に示したように，パケット取得時刻・パケットサイズ・宛先ポート番号の 3 つである．システムで利用するひとつの構造体が持つ情報はひとつのパケットの情報だけである．このようにすることで，情報が整理されていることから通信量算出部での処理が容易になるという利点がある．

3.2.4　表示機構

表示機構の構造を図 8 に示した．まず，表示機構の単位時間判別部において現在時刻を取得する．時刻差分算出にて初めて処理が渡されたときは，現在時刻からシステムを起動した時間の差を求める．求めた差が単位時間であった場合は，そのときの現在時刻を一時的に保存し，データ保存機構に単位時間分のデータを渡すよう要求を出し，処理が通信量算出部に渡る．%修士論文では時刻の取り方を図にする。求めた差が単位時間ではなかった場合，処理が現在時刻取得に戻る．2 回目からの時刻差分算出は，現在の時刻と一時的に保存した時刻との差を求める．

通信量算出部では，データ保存機構から渡されたデータから通信が存在した宛先ポート番号の通信量の総和を算出する．算出した通信量をソートにかけて，通信量の大きいものの上位 3 位までを決定する．ここまでの処理が終了したら，グラフ表示部に処理が移る．

グラフ表示部では，以上の一連の処理がなされた

表 2　動作実験の結果

		1	2	3	4	5	6	7	8	9	10
通信量1位	ポート番号	80	80	80	22	22	22	80	80	80	80
	通信量	27316	14769	10201	1504	1824	932	14338	26665	416	48601
通信量2位	ポート番号	22	22	22	0	53	0	22	22	22	443
	通信量	1212	1116	1932	0	343	0	1612	1108	92	38858
通信量3位	ポート番号	53	53	161	0	138	0	53	53	0	22
	通信量	610	249	105	0	229	0	336	115	0	9932

表 3　動作実験の結果

ネットワーク環境	無線 LAN (BUFFALO 製 WLI-CB-G54)
実験時間	10 分間
起動していたアプリケーション	ブラウザ(FireFox)
	ターミナルエミュレータ (TeraTerm Pro)

データの座標値変換を行う．座標値変換はデータをそのままの値で表示すると，ある単位時間の通信量が極端に多い場合は画面上における座標を超えた値になってしまう．それを防ぐために，ある一定の値を超えた場合は単位時間当たりの通信量の総和から 3 つの通信量の割合を算出し，変換後のデータをグラフ表示部において図 3 に示したようにグラフ化してユーザに表示する．同時に算出したデータを一時的に保存する．これは，過去のデータもグラフ化して一緒に表示するためである．さらに，ポート番号・プロトコル名・サービス名もグラフと共に表示する．プロトコル名・サービス名はあらかじめそれら一覧を示したテキストデータを用意しておき，ヘッダ情報の構造体データにおける宛先ポート番号から対応するものを抜き出し表示する．ユーザへの表示と同時に処理は再び単位時間判別部の現在時刻取得に戻る．

4.　動作実験

実装したシステムを実際に動作させる実験を行った．その動作実験の概要を表 3 に示した．2 種類のアプリケーションを使用しているが，これは 10 分間のあいだはずっと起動させておいた状態で断続的に使用した．このような状況で，動作させた結果が表 2 の動作実験の結果と図 9 のシステムのインターフェース画面である．表 2 は 1 分間ごとの通信量上位 3 位のポート番号とその通信量を表したものである．

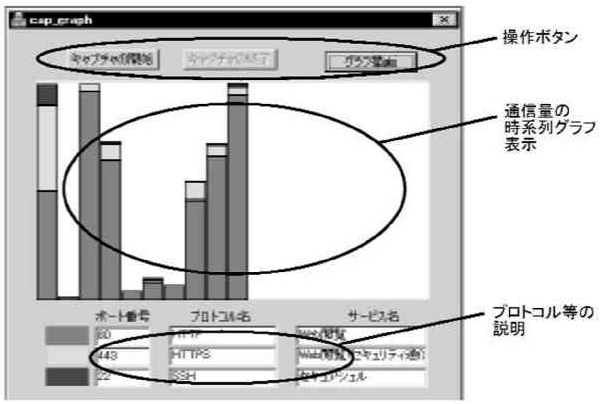

図 9　実際のシステムインターフェース画面

80 番ポートを使用するブラウザと 22 番ポートを使用するターミナルエミュレータを使用していたことから，このポートにおける通信がそのほとんどを占めていることがわかる．

図 9 からわかるように「キャプチャの開始」ボタンを押すことでパケットのキャプチャを開始し，「キャプチャの終了」ボタンを押すことでキャプチャを終了する．終了条件は，ボタンを押すことと設定した時間が終了することの 2 つである．また，今現在の段階では，リアルタイムに動作することができないので，設定した時間である 10 分経過後「グラフ描画」ボタンを押すことでグラフが表示されるという仕組みになっている．2.2 章でも述べたように，インターフェース画面の下側に表示されている情報は一番左側のブロックの情報，すなわち最も新しい通信量の情報を表している．よって図 9 では，最後の 1 分間の通信量上位 3 位のポート番号(80，443，22)，プロトコル名(HTTP，HTTPS，SSH)，サービス名(Web 閲覧，Web 閲覧(セキュリティ通信)，セキュアシェル)を表している．

5.　考察

動作実験の結果を表により表示した場合とグラフ

化して表示した場合とを比較すると，2.2 章で述べた，わかりやすさという観点からは明らかにグラフ化したほうが優れていることが見てとれる．例えば，図 9 からは実験開始時と終了時に通信が集中していることが一目で理解できる．よって，提案システムにより，ユーザに通信状況の概観を理解してもらうことはできるのではないかと考える．このことから，システムの設計方針で述べたパケットの可視化を実現できることを確認できた．

その他に動作実験からわかったこととして，システムにおける上位 3 位までの宛先ポート別通信量の表示が機能しない場合が 10 回中 3 回存在した．これは単位時間が短かったため，通信量自体が少なかったことが原因であると考えられる．この結果は今後のシステム改良に有益だと考える．

6. 今後の課題

本システムでは，わかりやすいインターフェースとパケットの可視化という 2 点をシステム設計の重要なポイントであると位置づけた．その中で，パケットの可視化は動作実験の結果からも達成できたと考えるが，インターフェースについて主観的に判断しても実装したシステムでは通信の概観は理解できても視覚的に興味をひくものとは言い難く，現状ではユーザに通信に興味を抱かせることは困難であると考える．よって，今後の課題としては，まずインターフェースの有効性を確認することが挙げられる．

また，今回実装したシステムでは宛先ポート番号・プロトコル名・サービス名の 3 つの表示であったが，システムの対象ユーザであるインターネット利用歴の低いユーザにとってわかりやすいものとは言い難い．加えて，宛先ポート番号が短命ポート番号の場合はプロトコル名が存在しない場合があるため，本システムではプロトコル名が表示されない可能性がある．以上 2 点を補うため，通信を行っているアプリケーションを宛先ポート番号などと同時に表示する機能を付加することも今後の課題である．

参考文献

[1] 独立行政法人情報処理推進機構: 情報セキュリティに関する新たな脅威に対する意識調査報告書 (2006).

[2] 小池英樹，高田哲司: 視覚表現による不正侵入検知システムの提案と実装，Cyber Security Magazine,1-1, pp.32-35 (2000).

[3] 斉藤匡人，金田裕剛，山下勝司，青柳禎矩，鵜飼文敏，徳田英幸: 3D-tcpdump: 通信トラフィックとネットワーク情報の視覚化ソフトウェア，日本ソフトウェア科学会 SPA 2005 ポスター (2005).

[4] Lloyd Treinich: Flow Visualization of Network Traffic, IEEE Computer Graphics and Applications September/October, pp.6-8 (1998).

[5] 荒井正之，田村尚也，渡辺博芳，小木曽千秋，武井惠雄: TCP/IP プロトコル学習ツールの開発と評価,情報処理学会論文誌，44-12，pp.3242-3251 (2003).

[6] 大橋正興，塚田浩二，安村通晃，小池英樹: Secure Sense:生活空間でセキュリティを「感じる」ための情報提示環境，情報処理学会シンポジウム論文集, 2003-7, pp.93-94 (2003).

[7] WinPcap:The Packet capture and network monitoring library for windows, http://www.winpcap.org/.

[8] 小高知宏: 基礎からわかる TCP/IP アナライザ作成とパケット解析 Linux/FreeBSD 対応，オーム社, 273 (2001).